N91-20706

ALLY: AN OPERATOR'S ASSOCIATE FOR SATELLITE GROUND CONTROL SYSTEMS

J. B. Bushman*, C. M. Mitchell, P. M. Jones, K. S. Rubin*
Center for Human-Machine
Systems Research School of Industrial and Systems Research
Georgia Institute of Technology
Atlanta, Georgia

Abstract

This paper explores the key characteristics of an intelligent advisory system. A central feature is that human-machine cooperation should be based on a metaphor of human-to-human cooperation. ALLY, a computer-based operator's associate is discussed which is based on a preliminary theory of human-to-human cooperation. ALLY assists the operator in carrying out the supervisory control functions for a simulated NASA ground control system. Experimental evaluation of ALLY indicates that operators using ALLY performed at least as well as they did when using a human associate, and in some cases they performed even better.

INTRODUCTION

Command and control (C2) systems have undergone dramatic changes within the last twenty years. Operators are faced with monitoring and controlling large, complex systems which rely heavily on the use of automaton. Often, the system is too large and complex for a single operator to monitor.

This paper presents the results of a research effort to explore the issues associated with human-machine cooperation in complex, dynamic supervisory control situations and to develop a theory of human-machine cooperation which can be used design the architecture for a computer-based operator's associate. The research focused on the development of a computer-based associate that is cable of cooperating with a human operator in monitoring and controlling a complex, dynamic system.

OPERATOR'S ASSOCIATE

As systems become more automated, the human operator performs fewer tasks on a routine basis. In complex dynamic systems, however, safety requires staffing at a level that can meet the most challenging or threatening abnormal conditions (Wickens, 1984). Normally, these worst-case conditions are well beyond the normal, day-to-day operational conditions. The result is often a team of human operators who are rarely challenged and often underutilized.

The concept of a computer-based operator's associate has been proposed as one method to remedy this situation and to provide intelligent decision aid for operators of complex dynamic systems (Chambers & Nagel, 1985; Rouse, Geddes, & Curry, 1987; Rubin, Jones, & Mitchell, 1988). An operator's associate is a computer-based system that acts as an assistant to the human operator. Functionally, an operator's associate can offer the operator timely advice and reminders, and at the operator's request, assume responsibility for portions of the supervisory control task.

The subordinate role of the operator's associate is a fundamental assumption that characterizes this research effort. The rationale for this assumption is that in complex dynamic systems it is impossible to anticipate and plan for all the contingencies. Thus, a computer system cannot act as the principal or sole "expert" in the system control; a human decision maker will always be present and ultimately responsible for effective and safe system operation. Thus, it is essential to design the system so that the operator is an integral part of the control and decision processes.

The intelligence and utility of the operator's associate rests on its abilities to understand the operator's current intentions and to provide context-sensitive assistance in the form of operator aids (e.g., suggestions, advice, and reminders) or by assuming responsibility for portions of the control task. To ensure generalizability, the operator's associate requires a well-defined knowledge structure. Knowledge concerning the controlled system, operator functions, and operator intentions must be represented (Chambers & Nagel, 1985; Rouse, et. al, 1987; Rubin et. al, 1988; Carroll & McKendree, 1987; Geddes, 1989; Hollangel, 1986; Sime & Coombs, 1983).

The understanding properties of the computer-based associate are based upon the existence of a model that prescribes the operator's interaction with the system (Rouse et. al, 1987; Rubin et. al, 1988; Geddes, 1989). Based on this model of the operator's actions, the automated associate must be able to monitor the operator's actions and model the current status of the decision maker

*Bushman is now with the Training Systems Division, Air Force Human Resources Laboratory, Brooks Air Force Base, Texas, 78235.

*Rubin is now with ParcPlace Systems, 1550 Plymouth St, Mountain View, California, 94043.

(i.e., intent inferencing) (Hollangel, 1986).

PRINCIPLES OF COOPERATION

The final property of a computer-based associate is that it should be based on the metaphor of human-to-human cooperation. The computer-based associate should interact with the human operator in a manner similar to the way in which humans interact in a cooperative environment (Carroll & McKendree, 1987; Hollangel, 1986; Fischhoff, 1986; Roth, Bennett & Woods, 1987; Woods, 1986a, 1986b; Woods, Roth & Bennett, 1987). An extensive empirical study was undertaken to investigate the nature of human-to-human cooperation that could serve as the basis for the architecture of an operator's associate.

The general principles of cooperation were derived from two sources. First, an extensive review of the literature was undertaken on cooperative problem solving. Second, extensive data was collected observing a team of experienced operators of the GT-MSOCC system (a typical cooperative supervisory control system) (Mitchell, 1987). The two operators were free to develop a "natural" style of interaction and cooperation. Verbal protocols were collected of the interactions between the operators and data describing their performance were collected. These protocols and data were then analyzed to describe the nature of their cooperative behavior.

A review of the literature indicated that a key principle of cooperation is that operators use multiple mental models to represent their knowledge of the physical system and their functions and to represent their knowledge of the other members of the cooperative team (Athans, 1982; Rasmussen, 1984, 1985; Tenney & Sandell, 1981a, 1981b). These distinct models serve to define and guide the interaction with the system and their interaction among the other operators.

The second feature of cooperation is referred to as cognitive balancing. This term is coined from the cognitive engineering approach to designing human-machine systems (Woods, 1986a, 1986b). Woods argues that the demands of the human and the system need to be considered and supported during the design of a human-machine system. With respect to a cooperative environment, the interacting operators must be aware of the cognitive demands and limitations of the other operators in order for efficient coordination and interaction to occur. One of the objectives of a cooperative team of problem solvers is to attempt to balance the joint cognitive demands of the team, as a whole. This balance is achieved through a mix of communication and delegation.

The final characteristic of cooperation is flexible levels of interaction. Empirical evidence supports the use of Rasmussen's levels of abstraction and aggregation (Rasmussen, 1984, 1985, 1986) to describe the content of the various mental models maintained by the operators and to describe the degree of interaction among the operators. The appropriate level of interaction is dynamic and is determined by the specific cooperation strategy. Interaction among the operators occurs at the levels of abstraction and aggregation common to the operators.

ALLY: A COMPUTER-BASED ASSOCIATE

These properties of a computer based associate and the principles of cooperation form the basis for the development of an architecture for a computer-based associate. The architecture is based on the OFMspert architecture (Rubin et.al, 1988. The architecture incorporates multiple models that represent the system knowledge, procedural knowledge, and operator intentions. The OFMspert architecture uses the operator function modeling (OFM) methodology as the basis for the design of an operator's associate. A key component of an operator's associate is the intent inferencing capability which provides the understanding properties for an intelligent operator's associate. The intent inferencing capability uses a blackboard architecture to understand the operator's current goals. The OFMspert intent inferencing capability was validated in Jones et. al (1989).

ALLY, a computer based associate, is based on an extension of the OFMspert architecture with control capabilities. The architecture provides an interface to the operator that allows the operator to retain complete control over the computer-based associate. The operator can delegate to the associate as many or a few of the tasks as desired.

ALLY was developed to assist an operator in carrying out the supervisory control function for a simulated NASA ground control system, called the Georgia Tech Multisatellite Operations Control Center (GT-MSOCC) (Mitchell 1987; Saisi, 1986). The design was based on a model of the GT-MSOCC operator control functions and attempts to duplicate the capabilities of a human associate. A detailed description of ALLY can be found in (Bushman, 1989).

The operational concept behind ALLY's design is that ALLY is based observations of the relationship that developed between a human operator and a human associate controlling the GT-MSOCC system. The human operator was in complete control of the human associate. The human associate, however, understood the cognitive complexities of the operator functions actively monitored the system for failures, and when necessary, would troubleshoot the system.

ALLY functions in a manner similar to the human associate. The operator has delegate as few or as many of the tasks to ALLY as desired. ALLY also actively monitors and troubleshoots the system on its own.

ALLY was developed in Smalltalk-80TM on a Macintosh II. ALLY interacts with the GT-MSOCC system in a distributed fashion. ALLY acts like another operator of GT-MSOCC system in a distributed fashion. ALLY acts like another operator of GT-MSOCC (see Figure 1). A distributed architecture is consistent with the concept of an assistant that executes autonomously and in its own environment.

Figure 2 provides an example of the ALLY interface to the operator. ALLY performs both delegated and automatic control tasks. The

TMSmalltalk-80 is a trademark of ParcPlace Systems, Inc.

operator delegates tasks to ALLY by selecting the corresponding control button. Each control button represents a specific operator control function as described in the GT-MSOCC operator function model (OFM) (Mitchell, 1987). Associated with each control button is a series of tasks that the human operator can delegate to ALLY.

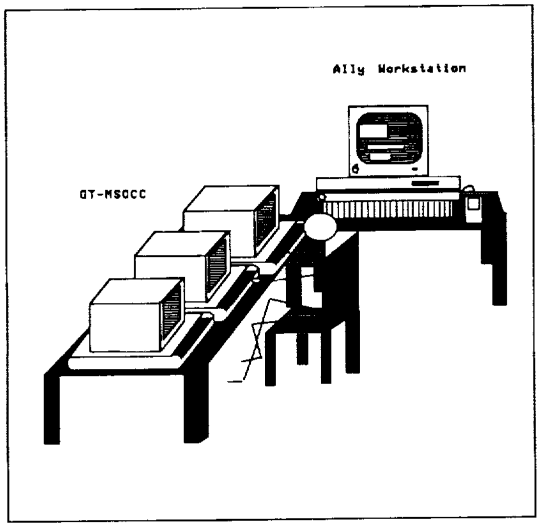

Figure 1. ALLY - GT-MSOCC Workstation

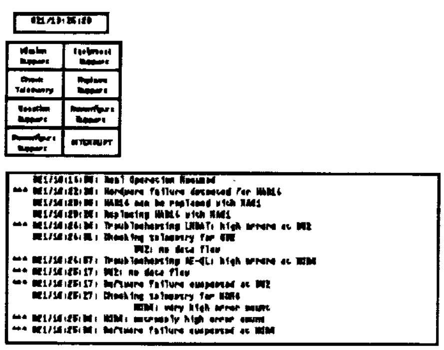

Figure 2. ALLY Basic Windows

The control buttons were designed with specific principles in mind. First, and foremost, the operator is provided the greatest degree of latitude to decide how much or how little support ALLY gives. The operator has complete control over the tasks ALLY performs. If the operator merely wants ALLY to determine the appropriate response and the operator wants to issue the various command, this level of support can be provided. On the other hand, if the operator wants ALLY to perform the entire function, this level of support is also accommodated.

While ALLY only performs the specific task assigned to it, it also understands the nature of the operator control functions. If ALLY knows that the function is still not complete, it offers to complete the task, if it can. It is important to note that this does not remove any of the control flexibility of the operator.

In addition to the delegated tasks, ALLY performs two tasks automatically. ALLY continuously monitors and troubleshoots the equipment networks. ALLY also automatically monitors critical events and offers reminders when it appears the the events might have been missed. This behavior is similar to that observed in a human associate working with the operator to control the GT-MSOCC system.

AN EXPERIMENT

An experiment was conducted to evaluate the effectiveness of ALLY as an operator's associate. The experiment compared the performance of an operator controlling GT-MSOCC working with ALLY as an associate with the performance of an operator working with a human associate.

Experimental Setup

The baseline GT-MSOCC system is a single operator system. In order to conduct the experiment, GT-MSOCC was modified to accommodate two operators. One operator serves as the primary operator and the second operator serves as an associate.

To support the associate position, two additional display screens were added to the baseline configuration. These two screens are functionally equivalent to the left and right screen in the baseline configuration. The center screen showing the GT-MSOCC Configuration and Status page is shared by the operator and associate. Although the physical display terminals are arranged in a different order, the functionality of the screens remain the same.

Each position is capable of issuing any of the GT-MSOCC operator control and information request command. Each position also has a dedicated audible alarm for system alarms. Common alarms indicating system events are sent to both positions, while operator error messages are only sent to the position which originated the error.

Subjects

Ten paid volunteer undergraduate Air Force ROTC cadets from Georgia Institute of Technology participated as subjects for the experiment. The subjects consisted of one female and nine males. The subjects included on junior, one sophomore, and eight freshman cadets. The subjects were paid six dollars per session.

Experimental Materials

Four sets of written instructions were used in the experiment. The first set consisted of an introduction to the baseline GT-MSOCC system and the operator supervisory control functions. These baseline instructions are found in Saisi (1986). The second set of instructions briefly described the operator-associate operations concept. The third set described the human associate concept and the modified GT-MSOCC workstation for a team of operators. Finally, the last set of instruction described the capabilities of ALLY and the user interface.

Several questionnaires were used during the experiment to collect subjective data. At the end of each data collection session, the subjects were asked to complete a Cooperation Evaluation carrying out the GT-MSOCC supervisory control functions. In addition, the subjects were asked to complete an ALLY Exit Questionnaire and a Human Exit Questionnaire at the end of their last data session with respective associate. The purpose of this these questionnaires was to elicit their opinions about various aspects of the associates. Finally, at the end of the experiment, the subjects were asked to complete a Subjective Comparison Rating questionnaire to compare their opinions about the two associates subjectively.

Overview of Experimental Sessions

The subjects were divided into two groups of five subjects each to control the order in which the subjects received the different associates. One group worked with the human associate first and the other group worked with ALLY first. In addition, to control for the variability of a human associate, a confederate was used in the experiment. The confederate was an expert GT-MSOCC operator and served as the human associate for each subject. The expert was instructed to use the same strategy for carrying out the operator control functions consistently to control the bias that might enter into the experiment from repeatedly seeing the same experimental sessions.

The subjects participated in twenty-four sessions: eight sessions of baseline GT-MSOCC training, three sessions of human associate training, four sessions human associate data collection, five sessions of ALLY training, and four sessions of ALLY data collection. A total of 240 hours of data was collected. The sessions were approximately 45 minutes in length. The sessions were run on consecutive days with typically one session per day. Occasionally, the subjects missed a day and made up the session by running multiple sessions in a single day.

Within each session, three hardware failures and six software failures were scheduled to occur. The failures were scheduled to occur at set times (as determined by the seed of a random number generator) on identical equipment across subjects for a given session. However, since all subjects did not operate the system identically, occasionally failures occurred on different pieces of equipment. In addition, three requests for support of unscheduled spacecraft contacts were also scheduled every session. Again, the sessions were structured such that the requests were identical across subjects for a given session.

Dependent Measures

Eleven baseline dependent measures were developed for GT-MSOCC (Mitchell & Saisi, 1987; Mitchell & Forren, 1987; Saisi, 1986). These measures plus five additional measures to determine how many of the different types of equipment failures were corrected by the subjects were used in the experiment. The performance measures are grouped into four categories: fault compensation, equipment configuration and deconfiguration, operator errors, and percentage of failures corrected.

The fault compensation measures reflect the time to compensate for each of the four types of failures. If the subject failed to compensate for the failure, the measure reflects the total time the failure was present in the system. The next group of performance measures reflect the time to respond to various equipment configuration and deconfiguration requests.

The operator error measures reflect the number of errors committed by the operator. Two types of errors can occur. The first type is when the operator causes a conflict with the automated scheduler. The second type occurs when the operator replaces a component that has not failed.

The last group of performance measures reflect the accuracy of the operator's fault detection strategy. The measure reflects the percentage of errors of a given type that the subject corrected during the session. A separate measure is used for each type of failure. In addition, a separate measure was used to reflect the percentage of total errors corrected.

Analysis

A mixed effect, nested factorial design was used to analyze the data. Because some of the dependent measures did not have a fixed number of repetitions per cell, the design was unbalanced in some cases.

The primary factor of interest is Condition which reflects the type of associate, i.e., human associate or ALLY. The experimental design was a repeated measures design in that each subject was exposed to both of the experimental conditions.

To control for the variability across the subjects, Subject was included as a factor in the experimental design. The Subject effect included 10 levels to reflect the 10 experimental subjects.

In order to account for any variability in the order in which the subjects worked with the two associates, Group was added as a factor in the experimental design. The Group factor includes two levels. The subjects in Group 1 worked with the human associate first, and the subjects in Group 2 worked with ALLY first. Subject, therefore, is a nested factor within Group.

Finally, Session was included as a factor to account for any variability between the sessions. The Session effect included four levels to reflect the four data collection sessions.

Analyses of variances were performed to determine the effect of each of the four independent variables (Condition, Group, Session, and Subject) on each of the sixteen dependent measures. An alpha lower-bar of .10 was used to detect significant effects.

Since the experimental design was a mixed design with random and fixed effects, approximate F statistics were constructed using Satterthwaite's method (Montgomery, 1984). Statistical analyses were performed using the General Linear Model (GLM) procedure of the SAS statistical software package (Spector, Goodnight, Sall, and Sarle, 1985). The GLM procedure computes the expected mean squares which were used to compute Satterthwaite's approximate F-statistic and the adjusted degrees of freedom. These values were then used to compute the significance level of the effects.

In addition to the statistical analysis, the results of the surveys and analysis of audit logs of the subjects' activities were examined to gain additional insight into the individual interaction questionnaire to capture subjectively the strategy they used to interact with the associate in strategies used by the subjects. These analyses, in conjunction with the statistical analyses, were used to evaluate the effectiveness of ALLY as an operator's associate and to evaluate the proposed theory of cooperation as it was implemented in ALLY.

DISCUSSION

The experimental results are summarized in Figure 3 and 4. Figure 3 summarizes the means and standard deviations for the two associate conditions across the 16 performance measures. Figure 16 provides a graphical comparison of ALLY's performance compared with the human associate. While these figures indicate that, on the average, ALLY tended to perform better than the human associate, only two of the performance measures yielded significant differences. These were the time to compensate for software type 1 failures (i.e., software failure characterized by termination of data flow) and the number of correct responses to unscheduled support requests. On all other performance measures ALLY performed as well as the human associate. A more exhaustive discussion of the results is found in Bushman (1989).

Dependent Measure	Human Associate		Ally		units
	Mean	Std. Dev	Mean	Std. Dev	
hardware failures	33.4	22.3	26.5	19.3	seconds
software failure 1	113.9	55.9	89.4	49.3	seconds
software failure 2	218.9	104.0	139.1	100.6	seconds
software failure 3	190.4	82.6	102.7	91.4	seconds
schedule conflicts	33.9	30.0	35.6	36.8	seconds
correct responses	2.3	0.7	2.8	0.5	per session
support requests	172.1	156.6	106.0	117.1	seconds
unscheduled contacts	165.3	151.6	120.5	174.6	seconds
deconfigure requests	7.6	5.7	8.7	11.6	seconds
operator error 1	1.2	0.9	0.9	0.8	per session
operator error 2	1.0	0.9	1.3	1.6	per session
% hardware fixed	99.2	5.3	100.0	0.0	percent
% software 1 fixed	83.7	23.7	92.5	18.1	percent
% software 2 fixed	85.0	25.8	93.7	20.2	percent
% software 3 fixed	91.2	19.2	98.7	7.9	percent
% total fixed	90.8	7.5	96.7	6.3	percent

Figure 3. Summary Performance Measures

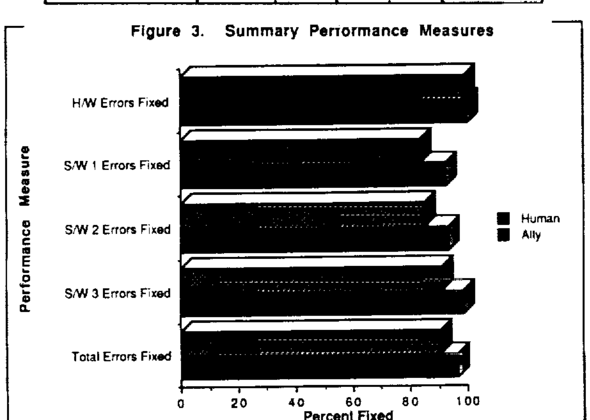

Figure 4c. Mean Performance Measures by Condition

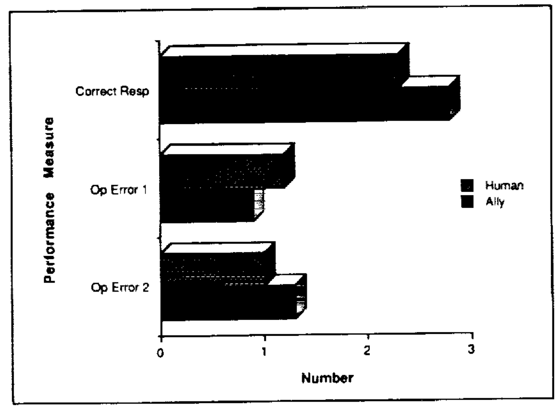

Figure 4b. Mean Performance Measures by Condition

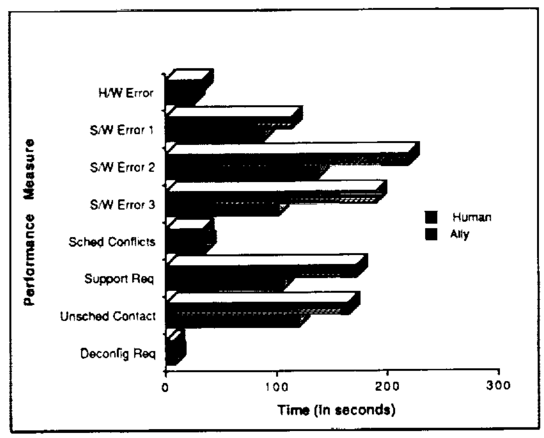

Figure 4a. Mean Performance Measures by Condition

While in only two cases a significant difference was detected between ALLY and the human associate, in most of the performance measures a significant condition by subject interaction was detected. This section presents the results of an in-depth analysis to attempt to explain these results.

Extensive audit records were recorded during each session of the experiment recording the behavior of the system, the behavior of ALLY, and the subject's interaction with both. These audit records were examined to investigate the reason for the significant differences among the subjects. The following sections present a discussion of the results in the four major categories of performance measures: fault compensation, equipment configuration, operator errors, and percentage of errors detected. Finally, the section concludes with a discussion of some of the subjective evaluations of the experiment derived from questionnaires.

Fault Compensation

The first category of performance measures reflects the time to detect and compensate for failures in the system. The analysis indicates that the effect ALLY had on performance depended

ORIGINAL PAGE IS OF POOR QUALITY

primarily on the cooperation strategy the subjects used. Subjects that used a more active strategy that takes advantage of ALLY's monitoring and troubleshooting control tasks were able to perform generally better with ALLY than with the human associate. Subjects that used a more passive strategy by relying on ALLY's automatic monitoring and troubleshooting capability, however, performed as well as with the human associate. Overall, the use of ALLY as an associate resulted in performance that was at least as effective as the human associate.

Equipment Configuration

The effectiveness of using ALLY as an associate in response to the various configuration and deconfiguration functions primarily is a factor of the subject's style of interaction. In responding to conflicts with the automated schedule, those subjects that chose to perform these tasks manually performed better than subjects that used ALLY. Lack of planning (ALLY cannot foresee these events) and the need to check ALLY's answers were the contributing factors to ALLY's slower performance.

ALLY performed as well as the human associate in responding to unscheduled support requests. ALLY, however, resulted in fewer incorrect responses than the human associate. No differences were detected with deconfiguration requests because the subjects performed most of these tasks manually, even when they had ALLY as an associate.

Operator Errors

The next category of performance measures relate to operator errors. Two types of errors were recorded. The first type of error relates to operator actions that cause a conflict with the automated schedule. The other type relates to replacing a component that had not failed.

With respect to the first type of errors (schedule conflicts), the analysis indicated that the subjects that used a more cautious strategy tended to generate fewer schedule conflicts. They would regularly check ALLY's replacements and the equipment it identified for support requests. The subjects that gave more responsibility to ALLY to replace components and schedule missions tended to generate more schedule conflicts.

No significant differences were detected with respect to the number of times the operator replaced a component that had not failed. This indicates that ALLY was just as effective as the human associate in correctly identifying equipment failures.

Percentage of Failures Detected

The analysis indicated that the subjects that used a more active fault compensation and detection strategy were able to detect more of the failures than the subjects that used a more passive strategy. The more successful subject consistently used ALLY to identify software failures before ALLY's automatic processing would detect them.

Subjective Evaluations

In addition to the above quantitative analysis, the subjects were asked to provide subjective evaluations of the two associates. Several different types of questionnaires were used to collect this information. This section summarizes the significant findings from these questionnaires.

In summary, the subjects felt that ALLY brought definite strengths to the task. ALLY's speed and accuracy at performing the monitoring tasks were cited as its major strengths. In addition, ALLY could quickly search schedules for free equipment.

On the other hand, they indicated several limitations to the use of ALLY. They had to build their trust in the system. Some of the subjects were able to build the confidence in ALLY and gave it more responsibility. Others, however, needed more experience with the associate before the trust could be established.

At times, ALLY was "resistive" in that it would not change its mind once it found an answer, but the subjects never felt like they were out of control because they had the capability to override ALLY's choices manually.

A common "fault" found with ALLY was that it made the job too easy. Those subjects that actively worked with ALLY to get it to do things, however, felt like they had more control over the situation because they were relieved from the mundane tasks.

Summary

Overall the performance of the subjects using ALLY as an associate was as effective as performance with the human associate. Individual strategies enabled some of the subjects to perform better with ALLY than with the human associate. The primary area that was affected by personal strategies was in detecting and compensating for software failures. Several subjects were able to develop a style of interacting with ALLY that enabled them to detect software failures before either one of them would on their own. This enabled them to detect the failures faster and to correct a larger percentage of the total failures.

Since ALLY does not have the capability to anticipate schedule conflicts, it is not able to plan for these events in advance. The subjects that relied on ALLY's capability to respond to these schedule conflicts could not take advantage of their planning ability. The subjects that performed the best with ALLY did not rely heavily on ALLY, but relied on their own capabilities to anticipate and plan for these events.

An unexpected result was a side-effect associated with the difficulty ALLY has with planning. ALLY performed as well as the human associate in responding to unscheduled support requests. However, because the subjects knew that this was one area in which ALLY can make mistakes, they regularly checked ALLY's answers. As a result, this additional checking resulted in more correct responses to support requests with ALLY.

Conclusions

This experiment demonstrated that a computer-based associate based on a model of the operator's function can perform as well as a human associate. As with any cognitive system (either human or artificial), ALLY brought with it strengths and limitations. The subjects that performed the best with ALLY were able to capitalize on its strengths and compensate for its weaknesses. The result was an overall increase in the system performance.

This research has demonstrated that a computer-based associate founded on the identified principles of human-machine cooperation can achieve performance compatible with a human associate. In addition, this research has provided a "starting-point" from which a finer theory of cooperation can be developed. The significance of this research is that it has provided empirical research concerning the nature of human-machine cooperation.

Quantitative experimental data demonstrated the feasibility of the architecture for a computer-based associate that can perform at least as well as a human associate. Qualitative data, in the form of subjective evaluations, identified some of the varied strategies used by operators to interact with a computer-based associate.

These quantitative and qualitative analyses may form the basis of a more refined theory of human-machine cooperation. Since no theory exists, this exploratory research is essential to develop a more definitive theory of cooperation.

ACKNOWLEDGMENTS

This research was supported in part by NASA Goddard Space Flight Center under contract NAS5-28575 (Walt Truszkowski, technical monitor), and NASA Ames Research Center NAG 2-413 (Everett Palmer, technical monitor).

REFERENCES

Athans, M., "The Expert Team of Experts Approach to Command-and-Control (C2) Organizations", IEEE Control Systems Magazine, Vol. 2, No. 3, 30-38, 1982.

Bushman, J. B., "Identification of an operator's associate model for cooperative supervisory control situations", Doctoral Dissertation, Report No. 89-1, Center for Human-Machine Systems Research, School of Industrial and Systems Engineering, Georgia Institute of Technology, Atlanta, Georgia, 1989.

Carroll, J. M. and J. McKendree, "Interface Design Issues for Advice-Giving Expert Systems", Communications of the ACM, Vol. 30, No. 1, 14-31, January 1987.

Chambers, A. B. and D. C. Nagel, "Pilots of the Future: Human or Computer", Communications of the ACM, Vol. 28, No. 11, 1187-1199, November 1985.

Fischhoff, B., "Decision Making in Complex Systems", in E. Hollnagel, G. Mancini, and D. D. Woods (Eds.), Intelligent Decision Support in Process Environments, 61-85. Berlin: Springer-Verlag, 1986.

Geddes, N. D., 1989, "Understanding Human Operator Intentions is Complex Systems", Unpublished doctoral dissertation, School of Industrial and Systems Engineering, Georgia Institute of Technology, Atlanta, Georgia, June 1989.

Hollnagel, E., "Cognitive System Performance Analysis", in E. Hollnagel, G. Mancini, and D. D. Woods (Eds.), Intelligent Decision Support in Process Environments, 211-226. Berlin: Springer-Verlag, 1986.

Jones, P. M., Mitchell, C. M. and Rubin, K. S., "Validation of intent inferencing by a model-based operator's associate", International Journal of Man-Machine Studies, 1989, in press.

Mitchell, C. M., "GT-MSOCC: A Research Domain for Modelling Human-Computer Interaction and Aiding Decision Making in Supervisory Control Systems", IEEE Transactions on Systems, Man, and Cybernetics, Vol. SMC-17, 553-570, 1987.

Mitchell, C. M. and M. G. Forren, "Multimodal User Input to Supervisory Control Systems: Voice-Augmented Keyboards", IEEE Transactions on Systems, Man, and Cybernetics, Vol. SMC-17, 594-607, 1987.

Mitchell, C. M. and D. L. Saisi, "Use of Model-Based Qualitative Icons and Adaptive Windows in Workstations for Supervisory Control Systems", IEEE Transactions on Systems, Man, and Cybernetics, Vol. SMC-17, 573-593, July 1987.

Montgomery, D. C. Design and Analysis of Experiments. New York: John Wiley & Sons, 1984.

Rasmussen, J., "Strategies for State Identification and Diagnosis in Supervisory Control Tasks, and design of Computer Based Support Systems", in W. B. Rouse (Ed.), Advances in Man-Machine Systems Research, Vol. 1, 139-193. Greenwich, CT: JAI Press, 1984.

Rasmussen, J., "The Role of Hierarchical Knowledge Representation in Decisionmaking and System Management", IEEE Transactions on Systems, Man, and Cybernetics, Vol. SMC-15, No. 2, 234-243, March/April 1985.

Rasmussen, J., Information Processing and Human-Machine Interaction: An Approach to Cognitive Engineering. New York: North-Holland, 1986.

Roth, E. M., K. B. Bennett, and D. D. Woods, "Human Interaction with an Intelligent Machine", International Journal of Man-Machine Studies, Vol. 27, 1-47, 1987.

Rouse, W. B., N. D. Geddes, and R. E. Curry, "An Architecture for Intelligent Interfaces: Outline of an Approach to Supporting Operators of Complex Systems", Human-Computer Interaction, Vol. 3, No. 2, 1987.

Rubin, K. S., P. M. Jones, and C. M. Mitchell, "OFMSPERT: Inference of Operator Intentions in Supervisory Control Using a Blackboard Architecture", IEEE Transactions on Systems, Man, and Cybernetics, Vol. SMC-18, 4, 618-637, 1988.

Saisi, D. L., "The Use of Model-Based, Window Display Interfaces in Real Time Supervisory Control Systems", Masters Thesis, School of Industrial and Systems Engineering, Georgia Institute of Technology, Atlanta, GA, 1986.

Sime, M. E. and M. J. Coombs, "Introduction", in M. E. Sime and M. J. Coombs (Eds.), Designing for Human Computer Communication, 1-20. London: Academic Press, 1983.

Spector, P. C., Goodnight, J. H., Sall J. P., and W. S. Sarle, "The GLM Procedure". SAS User's Guide: Statistics, Version 5 Edition, pp. 433-506. Gary, NC: SAS Institute Inc., 1985.

Tenney, R. R. and N. R. Sandell, "Structures for Distributed Decisionmaking", IEEE Transactions on Systems, Man, and Cybernetics, Vol. SMC-11, No. 8, 517-527, 1981a.

Tenney, R. R. and N. R. Sandell, Strategies for Distributed Decisionmaking", IEEE Transactions on Systems, Man, and Cybernetics, Vol. SMC-11, No. 8, 527-538, 1981b.

Wickens, C. D., Engineering Psychology and Human Performance. Columbus, OH: Charles Merrill, 1984.

Woods, D. D., "Cognitive Technologies: The Design of Joint Human-Machine Cognitive Systems", The AI Magazine, 86-92, 1986a.

Woods, D. D., "Paradigms for Intelligent Decision Support", in E. Hollnagel, G. Mancini, and D. D. Woods (Eds.), Intelligent Decision Support in Process Environments, 255-269. Berlin: Springer-Verlag, 1986b.

Woods, D. D., E. M. Roth, and K. Bennett, "Explorations in Joint Human-Machine Cognitive Systems", in A. Zachary and S. Robertson (Eds.), Cognition, Computing, and Cooperation. Norwood, NJ: Ablex, 1987.

*Bushman is now with the Training Systems Division, Air Force Human Resources Laboratory, Brooks Air Force Base, Texas, 78235.

*Rubin is now with ParcPlace Systems, 1550 Plymouth St, Mountain View, California, 94043.